APPENDIX A

Page 6, lines 5-9:

Next a quantization step 24 is ~~preformed~~performed on the wavelet data in which a uniform scalar quantization technique employs a dead zone around zero. According to the present invention, this technique prepares the structure of the wavelet data, allowing efficient compression. The dead zone at zero helps to increase compression without introducing large amounts of distortion. The compression level is adjusted by a user-controlled parameter CR that a[[e]]ffects quantization step 24.

a1

Page 6, lines 21-26:

The final step in the decompression procedure 40 is to inverse wavelet transform 46 the dequantized wavelet coefficients. Inverse wavelet transformation ~~46~~ step 46 produces the pixel values that are used to create the visual image. Some normalization of the pixel values is ~~preformed~~performed due to the rounding and quantization error involved in the forward wavelet transform step 22 and inverse wavelet transform step 46. The reconstructed image data 12 is then displayed to the user by any known image display hardware for digital data.

a2

Page 11, lines 17-21:

Referring to FIG. 2, dequantize step 44 is ~~preformed~~performed after step 42 of unpacking the data. First 0.5 is added to the quantized wavelet coefficient value. This shifting of the quantized value to the middle of the bin is used to prevent excessive rounding towards zero. This shifted value is then multiplied by user-defined parameter CR. In de-quantize step 44, if the quantized coefficient value is zero, then the value after dequantization will also be zero.

a3

Page 11, line 22 - page 12, line 1:

a4

In one embodiment of the invention, datapack step 26 includes several methods that are applied to different wavelet transform levels. FIG. 3 is a flow chart of an embodiment of a method of datapack step 26. At step 200~~S100~~, the image data is received. At step 202~~S102~~, adaptive run length coding is applied for three subbands (low-high, high-low, and high-high) at wavelet transform level one. At [[S]]step 204~~S104~~, for additional wavelet transform levels, a two-knob Huffman data pack method is performed. At step 206~~S106~~, a low frequency data packing method is used for the low frequency subband of the highest wavelet level. Each of the data packing methods will be described in detail herein.

Page 12, lines 7 - 18:

a5

At step 304, run-length coding on the zero coefficients is performed on the input data. The output of this run-length coding is a data stream consisting of non-zero coefficients[[,]] and zero plus run words. A run is herein defined as the number of zeros following the first zero of a sequence of sequential zeros. A sequence of just one zero is coded as a zero followed by a zero. A sequence of ten zeros[[,]] would result in a zero followed by a nine. Referring to FIGS. 9A and 9B, the input data sown in FIG. 9A includes three runs- run_1 having eight zeros, run_2 having 5 zeros, and run_3 having 10 zeros. As shown in FIG. 9B, the output data for run_1 includes a zero indicator and a run word of 7. The second run, or run_2, includes the zero indicators and a run word of 4. Shown in FIG. 12B is a representation of the data structure after run length coding in step 304. The zero indicator 524~~D24~~ is followed by run word 526~~D26~~. Run word 526~~D26~~ is followed by a non-zero

a5 cont'd

wavelet coefficient 528~~D28~~. Referring to step 306, shown in FIG. 8, the output data is further compressed using a Huffman table as illustrated in FIG. 10.

Page 13, lines 7-15:

a6

FIG. 12A illustrates a representation of the data structure after encoding a run in the present invention. There is a header element that provides information on the number of bits of the largest coefficient. Adjacent thereto may be a zero indicator element 510A~~D10~~ representing the~~that~~ start of~~to~~ a zero run. The next data objects 512, 514 and 516~~D12, D14, D16~~ are the components that make an exponential representation wherein the sum of 512, 514 and 516~~D12, D14, and D16~~ equals a power[[s]] of two. Thus, if a run length of 8 needs to be encoded then data objects 512, 514 and 516~~D12, D14, D16~~ would have a one therein and the sum of 512, 514 and 516~~D12, D14, D16~~ would be equal~~ed~~ to three. If the run length is greater than eight then the sequence is repeated. Zero indicator element 510B~~D10~~ is again encoded and data objects 518, 520 and 522~~D18, D20, D22~~ are encoded accordingly.

Page 15, lines 5-11:

a7

FIG. 17 illustrates test data of a chart of the Peak Signal-to-Noise Ratio (PSNR) values for different compression levels. The Peak Signal-to-Noise Ratio is the most commonly used metric of image quality in image and video compression literature. The test image was a high resolution picture suitable for publication in a magazine. The test image was 24-bit color, 1.86 x 2.79 inches at 1100 dots per inch (dpi). The data file size was 18.9 Mbytes. JPEG compression (illustrated as plot 702) was compared against the apparatus and methods in

a7 cont'd

accordance with the present invention (illustrated as plot 704). JPEG compression was limited to a compression level of 178:1[[,]]. A~~a~~s a result, the blocking effect was very noticeable and image quality was inferior to the present invention.